

Урок 1: Введение в Java



Преподаватель курса

Беланов Владимир

Преподаватель курса "Java Enterprise Developer"

Опыт работы:

Java разработчик Tinkoff.ru на проекте Tinkoff CRM.

Имею обширный опыт разработки высоконагруженных систем и управления командой разработки.

Суммарный опыт разработки: более 11 лет



Правила занятий

1

Занятие начинается ровно в 17:30. Опоздавших не ждем, опоздавшие тихо заходят не отвлекая всех остальных.

2

Вопросы можно и нужно задавать. Глупых вопросов не бывает.

3

Занятие длится 4 часа. Структура занятий: 2 часа лекция + 2 часа практика. Перерывы: 10 минут в середине лекции, 10 минут между лекцией и практикой, 10 минут в середине практического занятия.

4

Занятие это лекция+семинар. Самостоятельная работа дома. Отсутствующие отмечаются через 10-15 минут после начала занятия

Почему Java

- ✓ Количество вакансий Java разработчиков входит в ТОП-3 по России
- ✓ Размер оплаты труда Java разработчика входит в ТОП-3 по отрасли
- ✓ Java используется везде — от стартапов, до корпораций
- ✓ Java активно развивается. Новые версии выходят раз в пол года
- ✓ Огромное комьюнити

Спустя более 20-ти лет существования Java является одним из самых востребованных языков

Уровни разработчиков

Существуют следующие основные уровни разработчиков:

- Junior
- Middle
- Senior

Отличия:

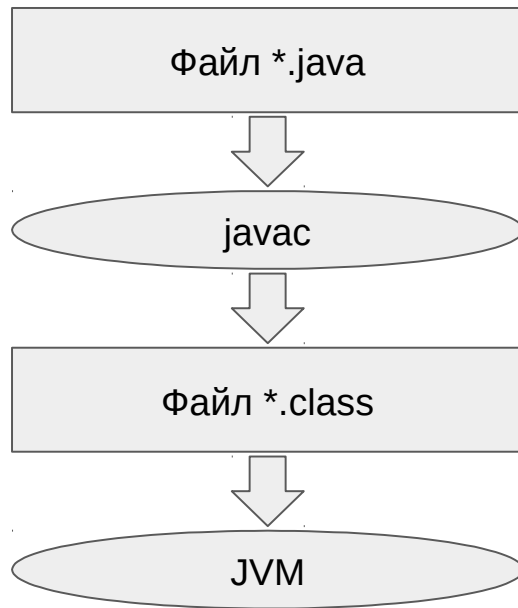
- может самостоятельно писать код и закрывать технические задачи;
- может проводить code-review;
- может самостоятельно уточнять требования системного и бизнес-анализа;
- обладает большой экспертизой в разных направлениях;
- может быть наставником;
- может самостоятельно принимать архитектурные решения
- предвидит проблемы;
- может распределять ресурсы внутри команды;
- может распределять ресурсы внутри проекта (несколько команд);

Путь долг и
предполагает
постоянное обучение

История Java

- 1991 – проект ОАК для IoT (Патрик Нотон, Джеймс Гослинг)
- 1992 – Средство интеллектуального дистанционного управления
- 1994 – разработка собственного браузера
- 1996 – первая версия языка java
- 1998 – Java 1.2
- 2004 – Java 1.5
- 2011 – Java 1.7
- 2014 – Java 1.8
- 2018 – Oracle объявила о смене принципа распространения ПО

Java из языка для
микроустройств
превратилась в
серверный язык



Особенности Java

- ✓ Объекто-ориентированность
- ✓ Строгая типизация
- ✓ Garbage Collector: Java не работает с памятью напрямую, поэтому объект создается, а потом очищаются ссылки на него. Остальное-ответственность GC
- ✓ Для исполнения кода требуется JVM
- ✓ Архитектурная независимость
- ✓ Встроенная многопоточность
- ✓ Встроенные коллекции

Виды Java-машин

Java SE

Standard edition -

Содержит базовые библиотеки, необходимые для использования языка:

- ✓ java.lang
- ✓ java.io
- ✓ java.math
- ✓ java.net
- ✓ java.util
- ✓ и т.д.

Java EE

Enterprise edition -

включает в себя инструменты для крупных отказоустойчивых многоуровневых и распределенных приложений, исполняемых на серверах приложений, такие как:

- ✓ JMS
- ✓ RMI
- ✓ JDBC
- ✓ JPA
- ✓ работа с XML и JSON
- ✓ сервлеты
- ✓ портлеты
- ✓ и т. д.

Java ME

Micro edition - специальное издание, созданное для использования в микроустройствах: телеприставках, системах умного дома и т. д. В настоящее время используется крайне редко.

IntelliJ Idea:

Ultimate edition

JDK:

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Выбираем подходящую версию JRE

Git:

<https://git-scm.com/download/win>

Выбираем подходящую версию клиента git

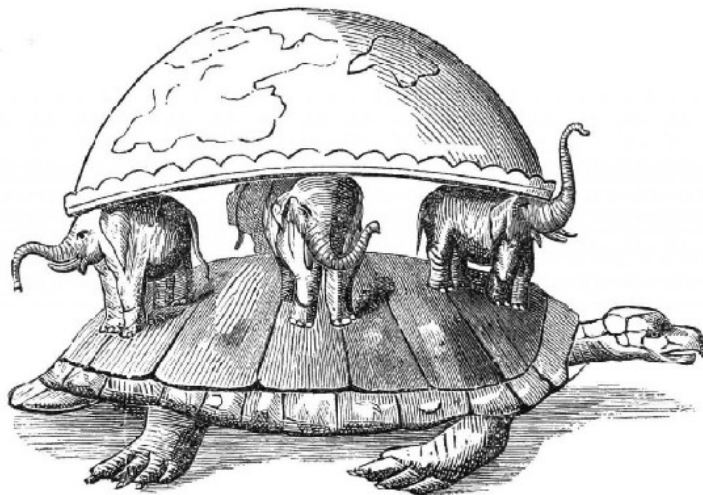
Object Oriented Programming

Объектно-ориентированное программирование - это шаблон проектирования ПО, позволяющий решать задачи разработчика с точки зрения взаимодействия объектов.

Объект — ключевое понятие в ООП.

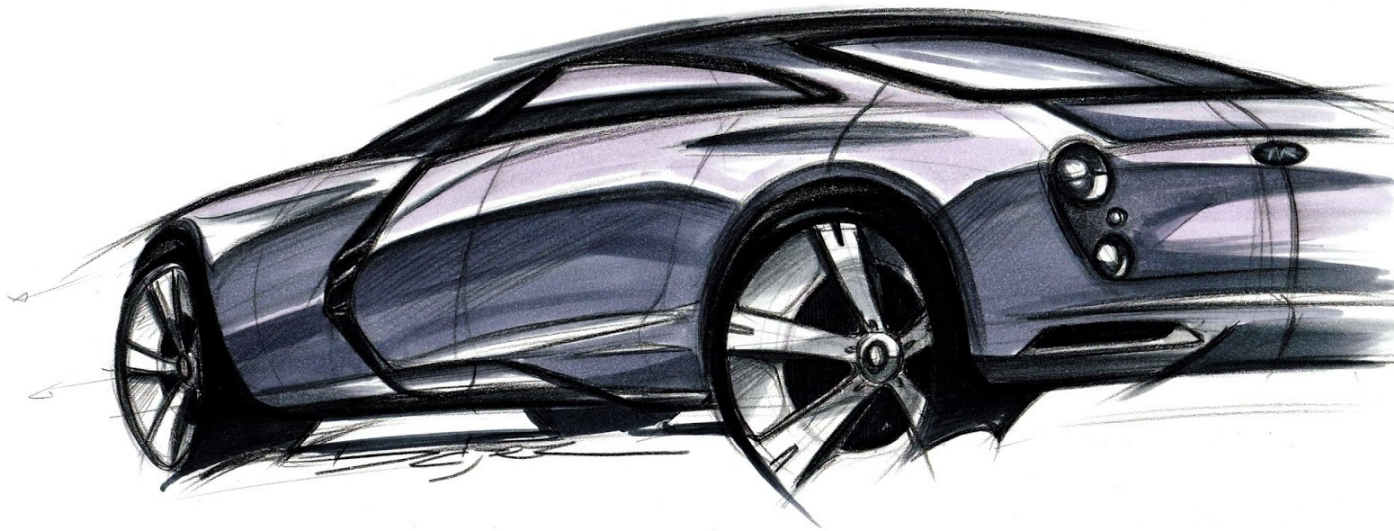
Столпы ООП:

- Абстракция
- Инкапсуляция
- Наследование
- Полиморфизм



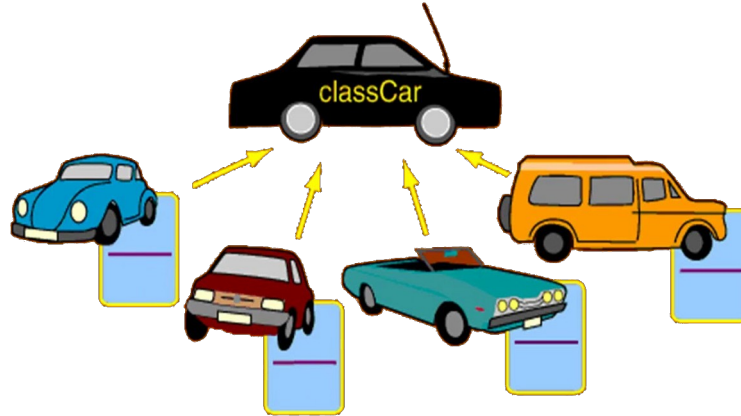
Абстракция

Абстрагирование - это способ выделить набор значимых характеристик объекта, исключая из рассмотрения не значимые. Соответственно, **абстракция** — это набор всех таких характеристик.



Наследование

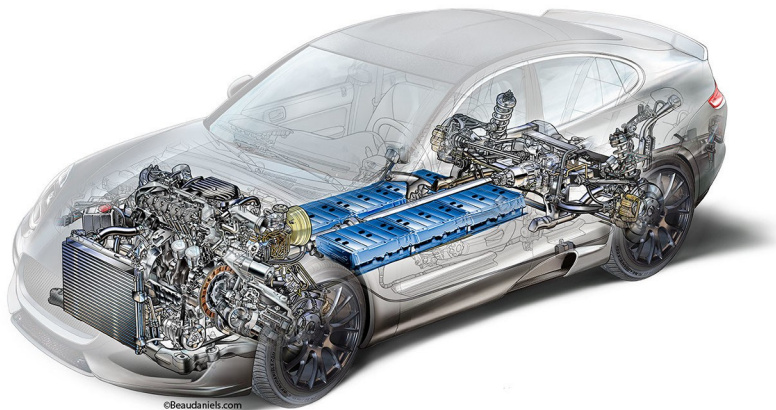
Наследование - механизм, позволяющий описать новый класс на основании родительского (существующего). Причём функциональность и свойства родительского класса заимствуются новым.



Инкапсуляция

Инкапсуляция — это свойство системы, позволяющее объединить данные и методы, работающие с ними в классе, и скрыть детали реализации от пользователя.

В Java инкапсуляция реализована в виде модификаторов доступа: **public**, **protected**, **private**, **default**.



Полиморфизм

Полиморфизм — это свойство системы использовать объекты с одинаковым интерфейсом без информации о типе и внутренней структуре объекта.

